CLAIMS

[0051] We claim:

1. A method comprising allocating spill cells used by an instrumentation fragment that has access to a single free register and that is run on a processor with a register stack architecture.

2. A method as in claim 1, where said allocating comprises:

designating an index for a spill array and a lock array;

incrementing said index;

loading said incremented index in said free register;

altering a value in a cell of said lock array;

determining whether said altered value in said cell of said lock array equals a pre-defined value; and

allocating said spill cell corresponding to said incremented index.

3. A method as in claim 2, further comprising reducing said incremented index by the number of cells in said lock array if said incremented index exceeds the number of cells in said lock array.

4. A method as in claim 2, wherein said incrementing said index comprises executing a threadsafe instruction.

5. A method as in claim 2, wherein said determining whether said altered value equals a pre-defined value comprises freeing a predicate register.

6. A method as in claim 2, wherein said altering said value in said lock array comprises incrementing said value.

7. A method as in claim 2, comprising reducing said altered value of said lock array if said altered value equals a maximum permitted value.

8. A method comprising:

storing an incremented index in a free register of a processor, such processor using a register stack architecture;

calculating in said free register the address of a cell of a first array corresponding to said incremented index;

loading in said free register an incremented value from said cell of said first array;

comparing said incremented value in said free register to a pre-defined value; and

allocating a cell of a second array corresponding to said index in said free register if said incremented value equals said predefined value.

9. A method as in claim 8, comprising reducing said incremented index modulo to the number of cells in said first array.

10. A method as in claim 8, comprising reducing said incremented value if said incremented value equals a maximum permitted value.

11. A method of spill cell allocation comprising:

storing an incremented value in a memory and in a free register of a processor that uses a register stack architecture;

comparing said incremented value in said free register to a pre-defined value;

allocating a spill cell if said incremented value in said free register equals said pre-defined value; and

re-setting said incremented value in said memory.

12. A method as in claim 11, comprising determining if said incremented value equals a maximum permitted value.

13. A method as in claim 11, comprising reducing said incremented value if said incremented value equals a maximum permitted value.

14. A device comprising a processor with a register stack architecture, said device capable of allocating a spill cell using one free register.

15. A device as in claim 14, said processor to:

store an incremented index of an array in said free register;

calculate in said free register the address of a cell of an array corresponding to said incremented index;

load in said free register an incremented value from said cell of said array;

compare said incremented value in said free register to a pre-defined value; and

allocate a spill cell of a spill array corresponding to said index in said free register if said incremented value equals said pre-defined value.

16. A device as in claim 15, said processor further to determine if said incremented value equals a maximum permitted value.

17. An article comprising a storage medium having stored thereon instructions that, when executed by a processor, result in: storing an incremented index of an array in a free register of a processor using a register stack architecture;

calculating in said free register the address of a cell of an array corresponding to said incremented index;

loading in said free register an incremented value from said cell of said array;

comparing said incremented value in said free register to a pre-defined value; and

allocating a spill cell of a spill array corresponding to said index in said free register if said incremented value equals said predefined value.

18. An article as in claim 17, wherein said instructions further result in determining if said incremented value equals a maximum permitted value.

19. An article as in claim 18, wherein said instructions further result in reducing said incremented value if said incremented value equals a maximum permitted value.

20. A system comprising:

a dynamic random access memory storage unit; and

a processor with a register stack architecture capable of allocating a spill cell using one free register.

21. A system as in claim 20, said processor to

store in said free register an incremented index of an array;

calculate in said free register the address of a cell of an array corresponding to said incremented index;

load in said free register an incremented value from said cell of said array;

compare said incremented value in said free register to a pre-defined value; and

allocate said spill cell of a spill array corresponding to said index in said free register if said incremented value equals said pre-defined value.

22. A system as in claim 20, said processor to determine if said incremented value equals a maximum permitted value.

23. A processor to:

store an incremented index in a free register of said processor, such processor using a register stack architecture;

calculate in said free register the address of a cell of a first array corresponding to said incremented index;

load in said free register an incremented value from said cell of said first array;

compare said incremented value in said free register to a pre-defined value; and

allocate a cell of a second array corresponding to said index in said free register if said incremented value equals said predefined value.

24. The processor of claim 23, the processor to reduce said incremented index modulo to the number of cells in said first array.

25. The processor of claim 23, the processor to reduce said incremented value if said incremented value equals a maximum permitted value.